



①⑨ BUNDESREPUBLIK  
DEUTSCHLAND



DEUTSCHES  
PATENTAMT

⑫ **Offenlegungsschrift**  
⑩ **DE 44 40 438 A 1**

⑤① Int. Cl.®:  
**G 06 F 11/25**

②① Aktenzeichen: P 44 40 438.7  
②② Anmeldetag: 11. 11. 94  
④③ Offenlegungstag: 15. 5. 96

DE 44 40 438 A 1

⑦① Anmelder:

Brutscher, Helmut, 82024 Taufkirchen, DE; Schult,  
Dagmar, 82024 Taufkirchen, DE; Piel, Werner, 85540  
Haar, DE; Birgmeier, Robert, 81739 München, DE

⑦② Erfinder:

Birgmeier, Robert, 81739 München, DE

⑤⑥ Für die Beurteilung der Patentfähigkeit  
in Betracht zu ziehende Druckschriften:

DE 40 00 943 C1  
DE 36 90 031 C2  
DE 43 20 351 A1  
DE 31 23 687 A1  
US 53 07 285  
US 52 53 255

Elektronik Informationen 12, 1993, S.38,40;

⑤④ Einrichtung und Verfahren zur Analyse von Rechnersoftware

DE 44 40 438 A 1

Die Erfindung betrifft eine Einrichtung und ein Verfahren zur Analyse von Rechner-Software, mit dem sich auf einfache Weise Software-Monitore und Software-Debugger für die Programmentwicklung auf eingebetteten Systemen ("Embedded Systems") um Funktionen wie z. B. Hardware-Breakpoints und Hardware-Trace ergänzen lassen.

Zum Test und zur Analyse von Software für "Embedded Systems" sind derzeit zwei verschiedene Verfahren im Einsatz. Bei dem einen Verfahren werden Hardware-Emulatoren verwendet, bei dem zweiten Verfahren wird die Testhilfe nur durch Software realisiert.

Bei Einsatz eines Hardware-Emulators werden die sich in einer Fassung auf der Zielhardware befindlichen Prozessoren, durch einen entsprechenden Adapter ersetzt, der über ein Kabel mit dem eigentlichen Emulator verbunden ist. Man erhält mit Hilfe dieser Hardware die Möglichkeit, die einzelnen Schritte der Befehlsabarbeitung des aus seiner Fassung entfernten Prozessors zu emulieren und alle Signale des Prozessors zu überwachen. Dadurch kann z. B. die Ausführung des zu testenden Programms bei einem bestimmten Signalmuster am Prozessor angehalten werden — nachfolgend wird dieser Fall "Hardware-Breakpoint" genannt — oder eine Historie von Signalmustern, die auszuführenden Einzelbefehlen entsprechen, aufgezeichnet werden — nachfolgend wird dieser Fall "Hardware-Trace" genannt. — Dadurch, daß alle Signale des Prozessors über den Adapter emuliert werden, können weitere Testhilfen, wie z. B. Emulationsspeicher hinzugefügt werden.

Für den Softwaretest auf eingebetteten System werden nun in zunehmenden Maße auch Software-Monitore und -Debugger anstelle von Emulatoren eingesetzt. Diese haben aber den Nachteil, daß nicht alle Eigenschaften eines Emulators, — wie z. B. die oben erwähnten Funktionen "Hardware-Breakpoints" und "Hardware-Trace" — ermöglichen.

Während Software-Monitore und -Debugger eine preisgünstige Lösung — mit den bereits erwähnten Nachteilen — für den Softwaretest auf eingebetteten Systemen darstellen, sind Emulatoren sehr teure Lösungen, die aufgrund ihrer physikalischen Verbindungen z. B. kurze Adapterkabel, und der Größe der Geräte oft nicht einsetzbar sind.

Der vorliegenden Erfindung liegt die Aufgabe zugrunde, die vorerwähnten Nachteile zu vermeiden. Ferner sollen Software-Monitore und -Debugger in einfacher Weise um die Funktionalität von "Hardware-Breakpoints" und "Hardware-Trace" erweitert werden, ohne dabei die erwähnten Nachteile eines Emulators aufzuweisen.

Erfindungsgemäß wird die Aufgabe durch eine Einrichtung zur Überwachung von Rechner-Software mit einem Trägerelement, mit einem Prozessor-Adapter und einem Prozessor-Sockel zur Aufnahme eines Prozessors und einer digitalen Schaltung zur Analyse von Signalen, die zwischen einem Prozessor, der auf den Prozessor-Sockel gesteckt wird und den Prozessor-Adapter, der auf eine Platine mit weiteren Schaltungen gesteckt werden kann, ausgetauscht werden, gelöst. Vorteilhafte Weiterbildungen sind in den Unteransprüchen beschrieben.

Die erfindungsmäßige Einrichtung ermöglicht die kontrollierte Ablaufunterbrechung eines Programms (Hardware-Breakpoints) in Mikroprozessorsystemen ohne Emulatoreinsatz und kann so für die Programm-

analyse und den Programmtest Emulatoren ersetzen.

Das Trägerelement wird anstelle des Mikroprozessors in ein Hardware-Zielsystem gesteckt. Das Zielsystem ist die Stelle, an der vorher der Prozessor in einer Platine oder dgl. angeordnet war. Der entfernte Prozessor wird dann in den Prozessor-Sockel des Trägerelementes eingesetzt. Auf dem Trägerelement werden nun mittels Überwachungs- und/oder Steuerschaltung kontinuierlich der Adress- und Datenbus des Zielsystems überwacht. Falls die Bedingungen für einen vom Benutzer definierten Hardware-Breakpoint erfüllt sind, wird durch das Trägerelement eine Unterbrechungsanforderung — im folgenden auch Interrupt genannt — ausgelöst, als wäre das Trägerelement ein Teil des Zielsystems. Falls die Programmunterbrechung durch Interrupts nicht gewünscht wird, können die Hardware Breakpoints auch nur angezeigt werden, z. B. über andere entsprechend an das Trägerelement angeschlossene geeignete Darstellungsmittel.

Die mit der Erfindung erzielbaren Vorteile bestehen insbesondere darin, daß das Trägerelement durch Ausbildung des Prozessor-Adapters, der dem Zielsystem-Interface zugeordnet ist und des Prozessor-Sockels, der dem Mikroprozessor-Interface zugeordnet ist, eine sehr kompakte digitale Bauweise der Software-Überwachungseinrichtung ermöglicht.

Ferner ermöglicht der modulare Schaltungsaufbau (Sandwich-Bauweise) eine einfache Erweiterung der Schaltung auf dem Trägerelement und dadurch weitere Funktionen durch die Integrationen separater Zwischenplatten. Außerdem existieren keine physikalischen Verbindungen zu externen Geräten über Kabel. Auch kann der direkte Einbau der erfindungsgemäßen Einrichtung in die Zielhardware ohne Beeinflussung der Funktion der Zielhardware oder des auszuführenden Programms realisiert werden. Durch die Kurzübertragungswege zwischen dem Prozessor und dem Zielsystem werden keine nachteiligen Laufzeiten generiert.

Schließlich erlaubt die Programmierbarkeit der Überwachungslogik an dem Trägerelement eine hohe Flexibilität durch austauschbare Programmlogik im Feld.

Auch kann ohne eine separate Spannungsversorgung die Software-Analyse durchgeführt werden.

Die erfindungsgemäße Lösung stellt eine "Plug and Play"-Lösung dar, bei der das Trägerelement lediglich an die Stelle des Prozessors in die Zielhardware und der Prozessor auf das Trägerelement gesteckt werden muß. Weiter sind keine Eingriffe weder in Hardware noch in Software des Zielsystems notwendig.

Die Möglichkeit der Überwachungseinrichtung als Erweiterung der vom Benutzer entwickelten Applikationen ist jederzeit möglich.

Das erfindungsgemäße System kann sehr leicht auf eine weitere Funktionalität erweitert werden, so daß das Trägerelement auf die speziellen Wünsche des Benutzers des Systems angepaßt werden kann.

Die mit der Erfindung erzielbaren Vorteile bestehen auch darin, daß statt des sehr teuren und komplexen Emulators lediglich ein Software-Debugger und der Software-Monitor vorhanden sein müssen, um die für den Software-Test auf dem Zielsystem — "Embedded System" — benötigten Funktionen "Setzen eines Hardware-Breakpoints" und/oder "Protokollieren des Hardware-Trace" für den Software-Test einsetzen zu können.

Nachfolgend wird die Erfindung eines zeichnerisch dargestellten Ausführungsbeispiel näher erläutert. In

der Zeichnung stellen dar:

Bild 1 Geräteaufbau,

Bild 2 Ein Blockschaltbild der digitalen Schaltungen des Trägerelements (Basis-System).

Bild 1 zeigt den Geräteaufbau des Basis-Systems, bestehend aus zwei zusammensteckbaren Platinen (1) und (2), die ein Trägerelement bilden. Die erste Platine ist die sogenannte "Zielsystem-Platine" während die zweite Platine die sogenannte "Prozessor-Platine" ist. Die Zielsystem-Platine weist einen Prozessor-Adapter (4) auf, welcher in eine Zielhardware einsteckbar ist. Die Prozessor-Platine weist einen Prozessor-Sockel (5) zur Aufnahme eines Prozessors (nicht dargestellt) auf. Ferner weist die Prozessor-Platine einen programmierbaren Baustein (7) zur Aufnahme einer Überwachungs- und Steuerschaltung (8) zur Analyse von Signalen bzw. Signalmustern auf. Weitere zwischen der Zielsystem-Platine und Prozessor-Platine einsetzbare Schaltungen sind auf einer oder mehreren Zwischenplatinen realisierbar und können einfach auf das Trägerelement gesteckt oder einsteckig mit diesem verbunden werden. Weitere Erweiterungen wie z. B. eine serielle Schnittstelle nach außen, eine Speichererweiterung für das Zielsystem oder Hardware-Trace-Einrichtungen sind durch das Einsetzen entsprechender Zwischenplatinen in das Trägerelement möglich.

Die Zielsystem-Platine enthält einen Mikroprozessor-Adapter zur Installation des Trägerelements in das Zielsystem der Hardware. Die Prozessor-Platine enthält einen Mikroprozessor-Sockel für den aus der Zielsystem-Hardware entnommenen Mikroprozessor. Auf der Unterseite der Prozessor-Platine befindet sich ein sogenanntes FPGA (Field Programmable Gate Array), das die gesamte Logik des in Bild 2 dargestellten Basis-Systems enthält.

Die in Bild 2 dargestellte Überwachungs- und Steuerschaltung (8) des Trägerelementes (3) besteht aus einem Mikroprozessor-Interface (15) welches mit dem Prozessor-Sockel (5) verbunden ist. Außerdem weist das Trägerelement (3) ein Zielsystem-Interface (16) auf dem Mikroprozessor-Adapter (1) auf. Über Datenleitungen bzw. Datenbusse (17, 18, 19) ist das Prozessor-Interface (15) mit dem Zielsystem-Interface (16) verbunden. Das Mikroprozessor-Interface weist eine Adresskonfiguration (21) auf, welche über einen Datenbus (20) mit einem Adress-Decoder (12) der Überwachungs- und Steuerschaltung (8) verbunden ist. Ferner ist der Adress-Decoder (12) mittels Steuerleitungen (22) und (23) mit einem Kontroll- und Breakpoint-Register (13) und mit einem Daten-Bestätigungsgenerator — Data Acknowledge Generator (14) — verbunden. Ferner ist eine Komparator- und Breakpoint-Erkennungslogikschaltung (9) vorgesehen, welche über Datenleitungen (25) und (26) mit den Datenleitungen (17) und (18) zwischen dem Zielsystem-Interface (16) und dem Mikroprozessor-Interface (15) verbunden ist. Die Komparator- und Breakpoint-Erkennungslogik (9) ist über eine weitere Datenleitung (24) mit dem Kontroll- und Breakpoint-Register verbunden und über Steuerleitungen (27) und (28) mit einer Unterbrechungsschaltung (Interrupt-Logik) (11) verbunden und mittels der Steuerleitung (10) mit dem Mikroprozessor-Interface (15) und über eine Steuerleitung (29) mit dem Daten-Bestätigungsgenerator (14) verbunden. Das Kontroll- und Breakpoint-Register (13) ist über Datenleitungen (33) und (34) mit den Datenleitungen (25) und (26) und über eine Datenleitung (30) mit der Unterbrechungsschaltung (11) verbunden. Die Unterbrechungsschaltung (11) ist über Datenleitungen (31)

und (32) mit dem Zielsystem-Interface (16) und dem Mikroprozessor-Interface (15) verbunden, — die voran beschriebene Überwachungs- und Steuerschaltung des Trägerelementes stellt das Basis-System der erfindungsgemäßen Einrichtung dar, welche über zusätzliche Platinen um weitere Funktionen erweitert werden kann.

Die erwähnte Adresskonfiguration (21) wird durch den Adress-Decoder ermöglicht und erlaubt die Zuordnung des Trägerelements bzw. der Schaltung des Trägerelements in einen für den Benutzer geeigneten Adressraum. In diesem Adressraum ist das Trägerelement wie ein gewöhnlicher Speicher (memory mapped I/O) vom Zielsystem und dessen Programmen ansprechbar.

Die Spannungsversorgungsconfiguration erlaubt die Auswahl einer Spannungsversorgung durch eine separate Quelle oder durch das Zielsystem. Durch die geringe Stromaufnahme des Basis-Systems ist für das Basis-System in den meisten Fällen die Spannungsversorgung des Zielsystems für den Mikroprozessor möglich. In diesem Fall kann das Trägerelement bzw. seine Schaltungen ohne Verkabelung und ohne zusätzliche Spannungsversorgung mit dem Zielsystem und Prozessor betrieben werden.

Alle Prozessorsignale des Zielsystems, mit der Ausnahme der Signale zur Interrupt-Steuerung durch die Interrupt-Logik und zur Bestätigung der Datenübernahme durch den Daten-Bestätigungsgenerator (Data Acknowledge Generator) werden direkt zur Prozessor-Platine und damit zum angeschlossenen Prozessor geführt und lediglich durch die Überwachungslogik beobachtet. Die nicht direkt zum Mikroprozessor geleiteten Signale werden durch die Software-Überwachungsschaltung (8) kontrolliert, wobei dazu das Kontroll- und Breakpoint-Register (13) und der damit verbundene Vergleich und Breakpoint-Erkennungslogik (9) dienen.

Die zu kontrollierenden Signale sind im wesentlichen:

- das Signal zur Bestätigung der Datenübernahme um Lese — oder Schreibzyklen abzuschließen,
- die Signale zur Anzeige einer Unterbrechungsanforderung, der vom Zielsystem oder vom Hardware-Breaker ausgelöst wurde,
- das/die Signal(e) zur weiteren Verarbeitung eines Interrupts.

Alle Signale, die von den Schaltungen des Trägerelementes überwacht oder verändert werden, sind gepuffert, so daß die Belastung des Mikroprozessors minimal ist.

Im operationellen Betrieb des Zielsystems — d. h. wenn die Schaltungen des Trägerelementes nicht zur Programmierung durch das Zielsystem angesprochen werden oder keine Unterbrechungsanforderung auslöst wird — werden diese Signale so schnell wie möglich durch die Trägerelement-Schaltungen geschleust (Basis-System). Die daraus resultierende Verzögerung von weniger als 10 ns hat gewöhnlich keinen Einfluß auf die Programmausführung des Zielsystems.

Ferner weist die Software-Untersuchungs- und Kontroll-Logik des Trägerelementes Anschlüsse für von außen anschließbare Steuerleitungen auf, durch die der Benutzer die Bedingungen zur Auslösung eines Hardware-Breakpoints bzw. zum Start des Hardware-Trace bei der Ausführung des zu testenden Programms definieren kann. Während der Ausführung des zu testenden Programms beobachten die Überwachungs- und Steuerschaltung des Trägerelementes die Daten auf dem

Daten- und Adressbus. Falls die Bedingungen für einen Hardware-Breakpoint erfüllt sind, wird eine Unterbrechungsanforderung ausgelöst bzw. ein Trace gestartet.

Durch die Unterbringung der kompletten Logik in einem FPGA, können insbesondere jederzeit andere/ neue Funktionen den Schaltungen zugeordnet werden, was mit einer sogenannten "im Feld Programmierung" möglich ist.

#### Patentansprüche

1. Einrichtung zur Überwachung von Rechner-Software, mit einem Trägerelement (3), das einen Mikroprozessor-Adapter (4) zum Einsetzen der Einrichtung in ein Hardware-Zielsystem, einen Prozessor-Sockel (5) zur Aufnahme eines Prozessors und eine Überwachungs- und Steuerschaltung (8), die zwischen den Prozessor-Adapter (4) und den Prozessor-Sockel (5) geschaltet ist, und zur Analyse von Signalen, die zwischen den Prozessor-Sockel (5) und dem Prozessor-Adapter (4) übertragen werden.
2. Einrichtung nach Anspruch 1, dadurch gekennzeichnet, daß das Trägerelement (3) zusammensteckbar ist und/oder Steckplätze für einsetzbare Zwischenplatten aufweist.
3. Einrichtung nach Anspruch 1 oder 2, dadurch gekennzeichnet, daß die Überwachungsschaltung (8) des Trägerelementes (3) einen Komparator (9) und eine Steuerleitung (10) zum Prozessor-Sockel (5) aufweist, und daß die Überwachungs- und Steuerschaltung (8) den Programmablauf in einem auf den Sockel gesteckten Prozessor unterbricht, wenn ein bestimmtes Signalmuster erkannt worden ist.
4. Einrichtung nach Anspruch 1 oder 2, dadurch gekennzeichnet, daß die Überwachungs- und Steuerschaltung (8) des Trägerelementes (3) Mittel zum Feststellen eines bestimmten Signalmusters in einem auf den Sockel gesteckten Prozessor aufweist.
5. Einrichtung nach Anspruch 4, dadurch gekennzeichnet, daß die Überwachungs- und Steuerschaltung (8) des Trägerelementes (3) eine Unterbrechungssteuerung (11) enthält, die die Ausführung des zu testenden Programms bei einem bestimmten festgelegten Signalmuster am Prozessor anhält.
6. Einrichtung nach Anspruch 4 oder 5, dadurch gekennzeichnet, daß die Schaltung des Trägerelementes (3) eine Aufzeichnungseinrichtung, z. B. Speicher für Signalmuster, auf einer Zwischenplatte ermöglicht.
7. Einrichtung nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, daß das Trägerelement (3) einen Signalausgang aufweist, an dem eine Wiedergabeeinheit zur Darstellung von Signalmustern anschließbar ist und/oder einen Signaleingang aufweist, über den mittels einer daran angeschlossenen Eingabeeinheit Bedingungen zur Programmunterbrechung in die Überwachungs- und Steuerschaltung des Trägerelementes (3) eingebbar sind.
8. Einrichtung nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, daß das Trägerelement (3) selbst Leitungen zur direkten Verbindung des Prozessor-Adapters (4) mit dem Prozessor-Sockel (5) aufweist.
9. Einrichtung nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, daß der Prozes-

sor-Sockel (5) über dem Prozessor-Adapter (4) liegt.

10. Einrichtung nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, daß das Trägerelement (3) Versteifungselemente aufweist, die an den Seiten des Trägerelementes (3) angeordnet sind.

11. Einrichtung nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, daß die Überwachungs- und Steuerschaltung (8) einen Adress-Decoder (11) aufweist, der Daten von der Adresskonfiguration des auf dem Prozessor-Sockel (5) gesteckten Prozessors empfängt und über Steuerleitungen mit einem Kontroll- und Breakpoint-Register (13) und/oder Daten-Bestätigungsgenerator (14) (Data Acknowledge Generator) auf dem Trägerelement (3) verbunden ist.

12. Einrichtung nach dem Anspruch 11, dadurch gekennzeichnet, daß das Kontroll- und Breakpoint-Register über Datenleitungen mit dem Zielsystem-Interface als auch mit einer Komparator- und Breakpoint-Erkennungslogik sowie mit der Unterbrechungssteuerung verbunden ist.

13. Einrichtung nach Anspruch 1, dadurch gekennzeichnet, daß die Überwachungs- und Steuerschaltung (8) des Trägerelementes (3) besteht aus:

- einem Prozessor-Interface (15), welches mit dem Prozessor-Sockel (5) verbunden ist;
- einem Zielsystem-Interface (16), welches mit dem Prozessor-Adapter (4) verbunden ist;
- Datenleitungen (17, 18, 19) zwischen dem Prozessor-Interface (15) und Zielsystem-Interface (16);
- einem Adress-Decoder (12), welcher über eine Datenleitung (20) mit einer Adresskonfiguration (21) des Mikroprozessor-Interfaces (15) verbunden ist und mittels einer Steuerleitung (23) mit einem Daten-Bestätigungsgenerator (14) verbunden ist;
- einer Komparator- und Breakpoint-Erkennungslogik (9), welche über Datenleitungen (17, 18) zwischen dem Zielsystem-Interface (16) und dem Mikroprozessor-Interface (15) verbunden ist, mit einer Datenleitung (24) mit dem Kontroll- und Breakpoint-Register verbunden ist, über Steuerleitungen (27, 28) mit einer Unterbrechungs-Schaltung (Interruplogik) (11) verbunden ist und mittels der Steuerschaltung (10) mit dem Mikroprozessor-Interface (15) und über eine Steuerschaltung (29) mit dem Daten-Bestätigungsgenerator (14) verbunden ist;
- das Kontroll- und Breakpoint-Register ist über Datenleitungen (33, 34) mit den Datenleitungen (25, 26) verbunden und über eine Datenleitung (30) mit der Unterbrechungsschaltung (11) verbunden;
- die Unterbrechungsschaltung (11) ist über Datenleitungen (31, 32) mit dem Zielsystem-Interface (16) und dem Mikroprozessor-Interface (15) verbunden.

14. Verfahren zur Analyse von Rechner-Software unter Verwendung der Einrichtung nach einem der vorhergehenden Ansprüche, gekennzeichnet durch folgende Schritte:

- a) Von einem Hardware-Zielsystem wird ein Mikroprozessor aus seiner Verbindung, vorzugsweise Steckverbindung, mit dem Zielsy-

stem gelöst;

b) der herausgelöste Prozessor wird auf den Prozessor-Sockel (5) des Trägerelementes (3) gesteckt;

c) mittels der Überwachungs- und Steuerschaltungen (8) auf dem Trägerelement (3) wird der Ablauf von Signalmustern in dem Prozessor erfaßt;

d) bei Vorliegen eines bestimmten Signalmusters wird der Programmablauf im Prozessor angehalten und/oder der Ablauf bestimmter Signalmuster wird in der Schaltung des Trägerelementes oder in einer darin angeschlossenen Aufzeichnungseinrichtung aufgezeichnet.

15. Verfahren nach Anspruch 14, bei dem die Schritte b) und c) vertauscht sind.

---

Hierzu 1 Seite(n) Zeichnungen

---

20

25

30

35

40

45

50

55

60

65

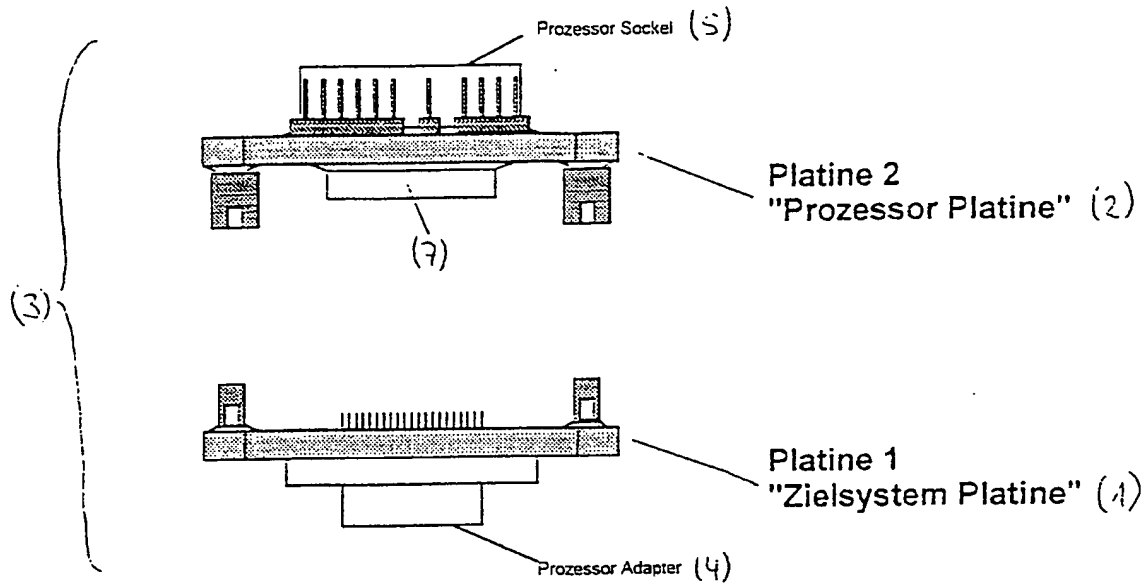


Bild 1: Geräteaufbau

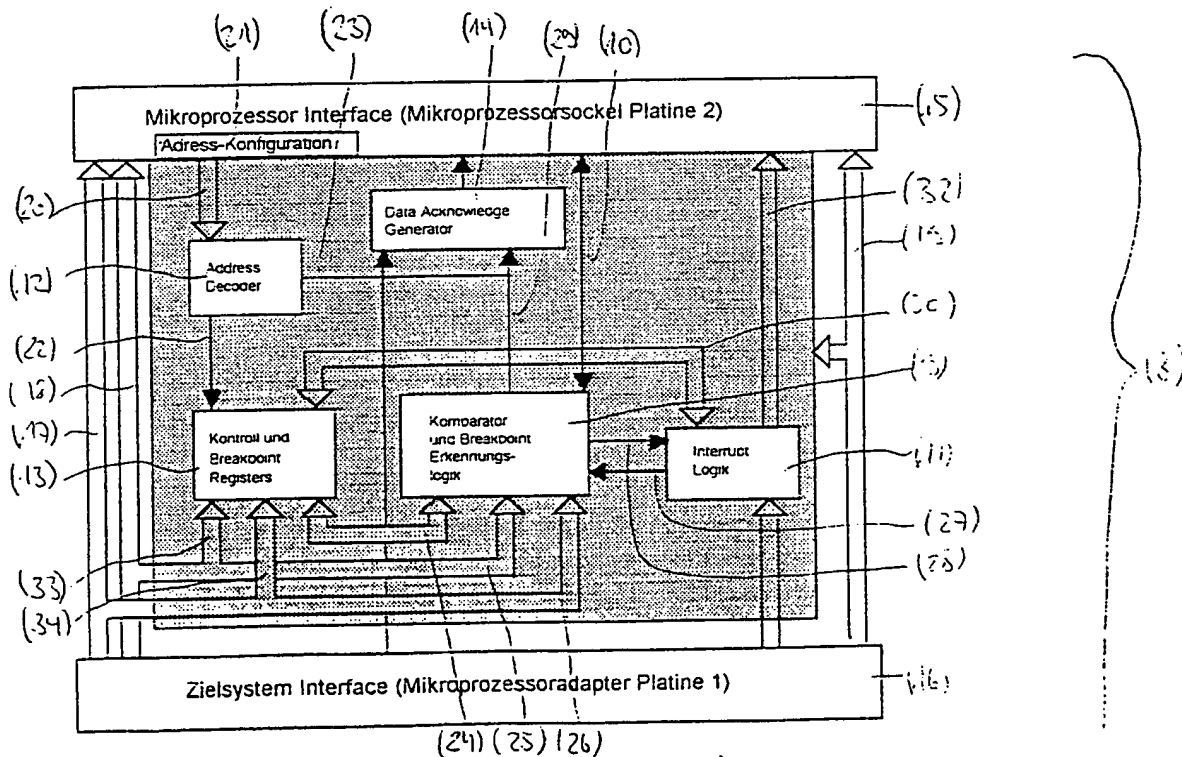


Bild 2: Hardbreaker-Logik (Basissystem)